ETHOS
AUDIT

ToastedAVAX

Audit Report

Apr. 29, 2022

Contents

Executive Summary 3
Audit Details 3
Methodology 3
Contract Details 4
Result Summary 4
Issues Reported 5
Issues Summary 5
Detailed Findings 5
TA-0 – Missing Event emissions 5
TA-1 – Functions that could be declared external 6
TA-2 – Centralization 6

Executive Summary

Audit Details

Project Name	Toasted AVAX
Codebase	https://snowtrace.io/address/0x1765e75bbF6cE8C43a13eD91C032A137d102f4d4#code
Initial Audit Date	April 29, 2022
Revision Dates	-
Methodology	Manual

Methodology

This audit's objectives are to evaluate:

- Security-related issues
- Code quality
- Relevant documentation
- Adherence to specifications
- Adherence to best practices

This audit examines the possibility of issues existing along the following vectors (but not limited to):

- Single & Cross-Function Reentrancy
- Front Running (Transaction Order Dependence)
- Timestamp dependence
- Integer Overflow and Underflow
- Mishandled exceptions and call stack limits
- Unsafe external calls
- Number rounding errors
- DoS with (Unexpected) Revert
- DoS with Block Gas Limit
- Insufficient gas griefing
- Forcibly sending native currency
- Logical oversights
- Access control
- Centralization of power
- Logic-Specification Contradiction
- Functionality duplication
- Malicious token minting

The code review conducted for this audit follows the following structure:

1. Review of specifications, documentation to assess smart contract functionality
2. Manual, line-by-line review of code
3. Code's adherence to functionality as presented by documentation
4. Automated tool-driven review of smart contract functionality
5. Assess adherence to best practices
6. Provide actionable recommendations

Contract Details

Contract ID	0x1765e75bbF6cE8C43a13eD91C032A137d102f4d4
Network	Avalanche C-Chain
Language	Solidity
Compiler	v0.8.13+commit.abaa5c0e
Verification Date	Apr. 27, 2022
Contract Type	Utility Contract
Libraries	OpenZeppelin

Result Summary

Ethos' audit of the ToastedAVAX smart contract has concluded with a **POSITIVE** result. The initial review identified a number of non-critical issues. The remaining report includes all issues identified in the initial review, as well as the revised status post resolution by the team.

- The smart contract is a variant of the 'miner' meta, with updates that allow for for additional flexibility for The Bakehouse team
- It allows users to deposit network native tokens into the contract
- Deposits are locked on deposit and redistributed to users over time
- The rate of redistributions approximately **5% daily** and varies based on the rate of increase of total value locked
- There is a referral bonus distributed to referrers of approximately **7%**
- There is a **3% dev fee** applied on all deposits and withdrawals
- The dev fee, daily reward and referral bonus rates are all updatable by the team through function calls. This would normally result in centralization concerns, however, since the values are constrained and the team is well known and successfull with the Baked Beans miner, full transparency with the community is expected and most likely a mitigating factor.
- Value locked within the contract cannot be manually removed by the owner
- The contract cannot be closed or shut off at any point after deployment

To conclude, this smart contract does what it is designed to, and is not ruggable by the owner or any other entities through attack vectors currently known in the EVM community.

Issues Reported

Severity	Unresolved	Acknowledged	Resolved
Extreme	0	0	0
High	0	0	0
Medium	0	0	0
Low	0	3	0

Issues Summary

ID	Title	Severity	Status
TA-0	Missing Event emissions	Low	Acknowledged
TA-1	Functions that could be declared external	Low	Acknowledged
TA-2	Centralization	Low	Acknowledged

Detailed Findings

TA-0 – Missing Event emissions

Severity: Low **Status**: Acknowledged

Description: There are several functions that change state variables, however, they do not emit events to pass the changes out of chain.

Risk: Not emitting an event from functions that impose changes to state variables could result in a lack of functionality often required for sound logic and functionality within external applications calling on smart contract functions.

Recommendation: We recommend emitting events for all essential state variables that are possible to be changed during runtime.

TA-1 – Functions that could be declared external

Severity: Low **Status**: Acknowledged

Description: Several functions are declared as public visibility, however, since they are never called by the contract they should be declared external.

Risk: This is a gas optimization issue.

Recommendation: We recommend that functions that are never called by the contract to be declared as external to save gas.

TA-2 – Centralization

Severity: Low **Status**: Acknowledged

Description: The dev fee, daily reward rate and referral bonus rates are updatable by the contract owner via function calls.

Risk: The team can potentially update these crucial miner values without being transparent with the community.

Recommendation: Since these functions are a feature that allows the Bake House team flexibility with the miner which gives it a better chance to adapt over a longer period of time, the only recommendation we can provide is that the Bake House team continue to stay fully transparent with the community on all contract changes.